Method for determining filter coefficients of a digital filter and digital filter

The invention relates to a method for modifying filter coefficients for a digital filter, more particularly for UMTS (Universal Mobile Telecommunication System), in which the filter coefficients are predetermined and modified in a filter design program. Furthermore, the invention relates to a digital filter of this type.

Digital filters are used, for example, as pulse shaping filters and matched filters in the UMTS standard (3GPP, TS 25.201: UE Radio Transmission and Reception (FDD), V3.2.0) and (3GPP, TS 25.213: Spreading and Modulation (FDD), V3.0.0). Owing to the number of filter coefficients to be processed and the large signal bandwidth such filters require much computation circuitry. Therefore, they can hardly be implemented by a signal processor which is part of the appliance that has the filter function. A hardware implementation necessary as a result and having its own components (chips) for the filters leads to much surface area in the appliance and to increased energy consumption. These two elements are thwarting a compact construction of the appliance.

In US-PS 6 311 203 is described a digital filter in which two factors are multiplied by to simplify the calculation of the filter coefficients. A preselection of filter coefficients that can be evaluated in a simple manner is not provided.

In US-PS 5 732 004 scaling factors are assigned to individual filter coefficients. A common scaling factor for all the filter coefficients is not provided.

From WO 01/22 582 A1 is known a floating point FIR filter. A common scaling factor is not provided.

It is an object of the invention to provide a method and a filter of the type defined in the opening paragraph in which without a decisive deterioration of the filter properties and of the processing rate, the filter is produced with a small chip surface and little current consumption.

The above object with respect to the method is achieved by the characteristic features of claim 1 and with respect to the filter by the characteristic features of claim 4.

For this purpose, the ideal filter coefficients previously determined in a filter design program are quantized and scaled so that they can be processed by a simple adding operation and costly multipliers are avoided. The filter coefficients simplified by quantization and scaling are not determined in the respective appliance that comprises the filter function. They are determined from the ideal filter coefficients in an extended filter design program and subsequently implemented in the appliance.

The quantized and scaled filter coefficients simplify the numerical complexity of the computation of the filtered signal. This allows to realize the filter function that has a small required area or space in the appliance and with little energy consumption. The filter may be an up-link and/or down-link, root-raised cosine filter of the UMTS with fixed-point arithmetic. The described measures, however, may also be used in other non-recursive filters working with fixed-point arithmetic.

Advantageous embodiments of the invention are apparent from the dependent claims and the following description. In the drawing:

Fig. 1 shows a FIR filter according to the state of the art,

Fig. 2 shows an improved filter arrangement with adder stages for the filter coefficients,

Fig. 3 shows one of the adder stages with three adders,

Fig. 4 alternatively shows an adder stage with one adder,

Fig. 5 alternatively shows an adder stage with selectable filter coefficients,

Fig. 6 shows a flow chart of the determination of the filter coefficients,

Fig. 7 shows a computation scheme for the multiplication by filter coefficients according to the state of the art,

Fig. 8 shows a computation scheme for the multiplication by a quantized filter coefficient,

Fig. 9 shows the quantity response for the evaluation of 4 "1" bits per coefficient,

Fig. 10 shows the quantity response for an evaluation of 3 "1" bits per coefficient,

Fig. 11 shows the quantity response for the evaluation of 2 "1" bits per coefficient,

Fig. 12 shows the quantization error in scaling factors between 1.5 and 2.3,

Fig. 13 shows a multiplier for the multiplication by squares.

In prior-art filters (Fig. 1) an input signal X is multiplied by filter coefficients b_{N-1} to b_0 in a plurality of multipliers 1. The filter coefficients are predetermined by means of a filter design program corresponding to the desired type of filter. The resulting partial signals Y_{N-1} to Y_0 are summed by adders 2 in a time-delayed manner (z^{-1}), the sum denoting the filtered output signal V. The ideal frequency response is shown in a dashed line in the Figs. 9 to 12 for a Root-Raised Cosine Filter. Fig. 7 shows an example for a multiplication according to the state of the art in which it is assumed that the signal value is decimal 217, thus 11011001 in binary form and the filter coefficient decimal 0.716796875 is 0.101101111 in binary form. For a random coefficient with nine binary fractional digits, eight computation steps are necessary. This leads to a high filter complexity.

In the described filter, coefficients which are simplified by scaling and quantization are used which can be processed in a simpler manner. The flow chart of Fig. 6 diagrammatically shows how new filter coefficients β_v are formed from the predetermined filter coefficients b_v. This is effected prior to the implementation of the filter coefficients in the filter.

Once the filter coefficients b_v typically of a filter have first been determined by a standard filter design, they are divided by a scaling factor which is the same for all the filter coefficients. The result is the scaled filter coefficients $\beta_v = b_v/s$.

Subsequently, a quantization of the scaled filter coefficients takes place. During the quantization the number n of the "1" bits found after the most significant bit (MSB) is limited to a certain maximum number n, where n is 4, 3 or 2 in the examples described. The number n may be different for each adder stage depending on the coefficients provided. This is based on the recognition that a multiplication of a signal value by a filter coefficient having few "1" bits needs fewer adding operations than a multiplication by an arbitrary filter coefficient of a same effective word length.

As a result of the quantization, distortions of the filtered signal will inevitably occur. This quantization error is smallest possible.

After the quantization the respective quantization error:

$E(s) = \Sigma|b_v - s\Sigma\ \beta_v|^2$

is determined (compare Fig. 6) and stored. Then the scaling factor s which was, for example, 1.0, is increased by one step (s_{step}) and the quantization error occurring with this new scaling factor is again determined and stored. If then the scaling factor increased in steps exceeds 2.0, a comparison is made what scaling factor s_0 leads to a minimum quantization error. This scaling factor s_0 is then used for the final determination of the filter coefficient β_v found via scaling and quantization, where

$\beta_v = Q(b_v/s_0)$

Q then describes the quantization.

At the output of the filter the division of the original filter coefficient by the common value s_0 is canceled as a result of a multiplication by the common scaling factor s_0.

The following Table shows stages of the quantization algorithm in the light of a numerical example:

Stage	Binary	Decimal	quantization error
original value	0.101101111	0.716796875	-
step 1: 4 "1" bits	0.101101	0.703125	0.013671875
consider next bit	0.1011011	0.7109375	0.005859375
round at fourth "1" bit	0.101110	0.71875	0.00195125

In the Table it is assumed that the number n of "1" bits allowed after the most significant bit is n = 4.

As appears from the Table the binary original value has seven "1" bits. In accordance with the guideline, in step 1 the binary value is limited to four "1" bits after the most significant bit, here 0. This corresponds to a decimal value of 0.703125, which as against the decimal original value means a quantization error of 0.013671875.

In a second step the next bit following the last "1" bit of step 1 is considered, which again is a "1" bit in the example. Since this bit is a "1" bit, a rounding is made in the next step, so that the binary value 0.1011110 arises. This corresponds to a decimal value 0.71875 which, compared to the decimal original value – irrespective of the scaling factor used – means a quantization error of –0.00195125. The rounding thus considerably reduces the quantization error.

Fig. 8 represents in comparison to Fig. 7 the computation scheme in which "217" is processed with said quantized and scaled filter factor in a decimal way: 0.71875 or

binary way 0.101110, respectively. Based on the quantization scheme used the filter factor can be unambiguously described by the (maximum n) positions of all bits set to "1", in this case by the positions i_0, k_0, l_0 and m_0 having the values 1, 3, 4 and 5. This processing requires only four steps i_0, k_0, l_0, m_0. Each multiplication by this „1" corresponds to a shift of the input signal, here decimal: 217, binary 11011001. Accordingly, i_0 is a shift by one position, k_0 is a shift by three positions, l_0 a shift by four positions, m_0 a shift by five positions. Such shifts can be realized with little circuitry because only data lines need to be properly connected for this purpose, as is shown by Fig. 13 via a shift by one position.

As a result of the described quantization, the optimum scaling factor s_0 can be sought in a limited interval, for example, in the interval from 1 to 2 or in the interval from $1/2\sqrt{2}$ to $\sqrt{2}$.

Said filter coefficients $ß_v$ allow a simplified filter structure or a simplified implementation of the filter, respectively, because multipliers are replaced by adders.

Fig. 2 schematically shows the structure of the filter. The multipliers 1 of Fig. 1 are replaced by adder stages ADD3 in which the scaled and quantized filter coefficients $ß_v$, $ß_{N-1}$ to $ß_0$ respectively, are processed. In a final stage 4, which may be arranged as a multiplier, a sum signal is processed with a factor s_0 reciprocal to the scaling factor of the filter coefficient $1/s_0$, to do away with the effect the scaling factor has on the output signal V. The final stage 4 may be omitted if the absolute value does not play a decisive role.

Fig. 3 shows one of the adder stages ADD 3 for the filter coefficient b_{N-1}. Fig. 3 relates to the case where said number n = 4, thus the filter coefficients are limited to four "1" bits per coefficient. Accordingly, four multipliers 5, 6, 7, 8 are provided which are designed for the multiplication by squares, that is, $-i_{N-1}$, $-k_{N-1}$, $-l_{N-1}$, $-m_{N-1}$ in accordance with the respective filter coefficient, while these powers correspond to the example i_0, k_0, l_0, m_0 of Fig. 8.

The result of the multipliers 5 and 6 is summed in an adder 9. The result of the multipliers 7 and 8 is summed in an adder 11. The results of the two adders are added in a further adder 10 to an intermediate signal Y_{N-1}. The number of adders 9, 10, 11 is n-1 in this case and in the other examples of embodiment. If n = 1, only one multiplier and no adder is necessary.

In the example of embodiment shown in Fig. 4 it is assumed that n = 2, thus the filter coefficients are limited to two "1" bits. Accordingly, only two multipliers 5, 6 and one adder 10 are necessary.

In the example of embodiment shown in Fig. 5 the adder stage 3 comprises a programmable selector 12. Due to the programmable structure the filter coefficients need not of necessity be determined prior to the implementation of the filter. Rather a general programmable filter can be implemented with the selectors, the filter coefficients of which have a maximum of n "1" bits. It is again assumed here that n = 4 (compare fig. 8), so three adders 9, 10, 11 are necessary. By means of the programmable selector 12 its outputs X_i, X_k, X_l, X_m are selected accordingly. The range of values of the inputs i, k, l and m is situated between 0 and M-1 and connects one of the respective inputs X_0 ... X_{M-1} with one of the outputs. The output X_i is determined by i, x_m is determined by m and so on. For example, a programming of the inputs with i = 1, k = 3, l = 4 and m = 5 selects the outputs at $X_i = X_1$, $X_k = X_3$, $X_l = X_4$ and $X_m = X_5$. M in this example is 8 (compare Fig. 7, Fig. 8).

Fig. 13 shows one of the squaring multipliers 5, 6, 7, 8. This multiplier is formed by a simple connection of the data lines between the 8-bit input and the 8-bit output. Fig. 13 shows the connection for the case of the shift by one position. In the other cases mentioned a respective other connection is selected.

Fig. 8 represents the quantity response which appears in a filter having filter coefficients selected in the above way in unbroken lines, where n = 4. The comparison of the solid line with the dashed line shows that there is no considerable deviation as a result of said selection of filter coefficients.

Fig. 10 represents the quantity response with filter coefficients with n = 3 in solid lines. The comparison shows that from about 4 MHz there are slight deviations. Accordingly, Fig. 11 represents the quantity response for filter coefficients with n = 2. Here too there are certain deviations only starting from 3.5 MHz.

Fig. 12 shows the curve of the quantization error E(s) described above with scaling factors in the range from 1.6 to 2.1. The quantization error shows a minimum at about 1.97. This value is accordingly used as the scaling factor s_0.

In all examples of embodiment the quantization and optimization of the scaling factor has achieved that instead of costly multipliers (Fig. 1), simple adder stages 3 can be used which reduces the required chip area of the hardware. The multipliers 5 to 8 of the adder stage 3 can be manufactured in a considerably simpler fashion, contrary to the multipliers of Fig. 1. Since fewer arithmetic operations are necessary than in the state of the art, also the power consumption is reduced. The distortions of the signal are slight and depend on the selection of the number n. The processing rate is increased if the additions in all adder stages 3 are carried out simultaneously and by equally many adders.

When the proposed quantization is used in a UMTS Root-Raised Cosine Filter, it is possible in a filter with symmetrical coefficients of length 32 to replace the 16 12-bit multipliers 1 with 16 12-bit adder stages 3.